Spherical Orbifold Tutte Embeddings

NOAM AIGERMAN, Weizmann Institute of Science
SHAHAR Z. KOVALSKY, Duke University
YARON LIPMAN, Weizmann Institute of Science

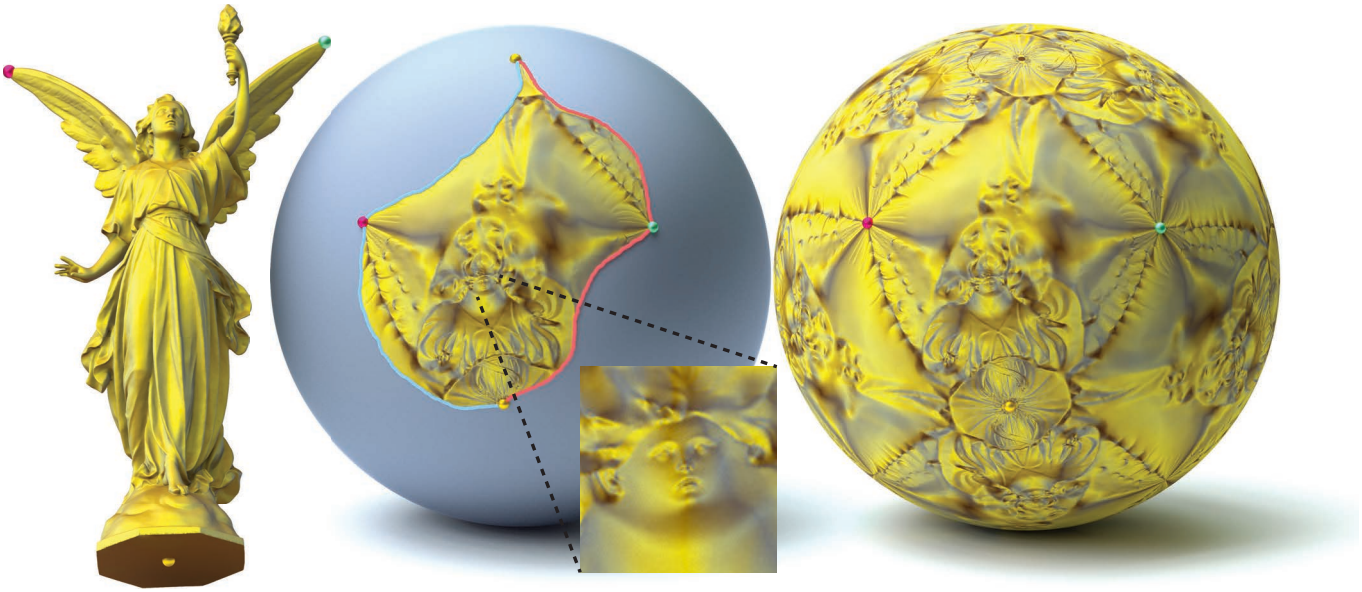


Fig. 1. A spherical orbifold Tutte embedding: a mesh (left) is embedded to a tile on the sphere (middle), which generates a seamless tiling of the entire sphere (right). Three points (visualized as colored balls) are designated as cones – symmetry points of the tiling.

This work presents an algorithm for injectively parameterizing surfaces into spherical target domains called spherical orbifolds. Spherical orbifolds are cone surfaces that are generated from symmetry groups of the sphere. The surface is mapped the spherical orbifold via an extension of Tutte's embedding. This embedding is proven to be bijective under mild additional assumptions, which hold in all experiments performed.

This work also completes the adaptation of Tutte's embedding to orbifolds of the three classic geometries – Euclidean, hyperbolic and spherical – where the first two were recently addressed.

The spherical orbifold embeddings approximate conformal maps and require relatively low computational times. The constant positive curvature of the spherical orbifolds, along with the flexibility of their cone angles, enables producing embeddings with lower isometric distortion compared to their Euclidean counterparts, a fact that makes spherical orbifolds a natural candidate for surface parameterization.

CCS Concepts: • **Computing methodologies → Computer graphics**;

Additional Key Words and Phrases: spherical parameterization, orbifolds, Tutte's embedding

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2017 ACM. 0730-0301/2017/7-ART90 $15.00
DOI: http://dx.doi.org/10.1145/3072959.3073615

ACM Reference format:
Noam Aigerman, Shahar Z. Kovalsky, and Yaron Lipman. 2017. Spherical Orbifold Tutte Embeddings. *ACM Trans. Graph.* 36, 4, Article 90 (July 2017), 13 pages.
DOI: http://dx.doi.org/10.1145/3072959.3073615

1 INTRODUCTION

This work lies at the connection between two different mesh parameterization methods: spherical parameterizations, and cone-manifold parameterizations.

The sphere is one of the most natural target domains for embeddings of genus-zero surfaces. However, parameterizing surfaces into the sphere is a non-trivial task as its non-convexity poses a challenge when used as a target domain, leading to numerous different approaches to tackling this problem, [Crane et al. 2013; Friedel et al. 2007; Gotsman et al. 2003; Kazhdan et al. 2012].

Parameterizing surfaces into planar *cone manifolds* has also been steadily gaining popularity, as it is a practical and effective way to embed closed (*i.e.*, boundaryless) surfaces into planar domains in a seamless manner, and the addition of cone singularities significantly reduces the distortion of the parameterization [Myles and Zorin 2012; Springborn et al. 2008]. As far as we are aware, in computer graphics cone singularities have yet to be used in the spherical setting.

In this paper we combine the two approaches, and introduce a simple and efficient algorithm for spherical cone parameterization. Theoretically, the algorithm is proven to produce valid, injective embeddings assuming a couple of additional conditions hold, and practically, in all conducted experiments produced such valid embeddings. The algorithm is based on an extension of the classic Tutte embedding [Floater 2003; Tutte 1963] to the spherical setting, with the *spherical orbifolds* as target domains. The spherical orbifolds are a collection of constant-positive-curvature surfaces with cone singularities; each spherical orbifold is generated from a symmetry group of the sphere.

The motivation to combine orbifolds and spherical embeddings is that the special properties of the spherical orbifolds, when combined with a spherical version of Tutte's algorithm, are shown to alleviate past problems of spherical parameterizations such as degeneration of the map (to a point or a line), multiple wrapping of the sphere, inverted elements, and "tent" singularities. Furthermore, the cone-singularity structure enables both setting 2-3 point constraints in the embedding, and also approximating conformal maps to spherical domains while reducing scaling near the cones. Figure 1 shows an example of a mapping of a mesh with sphere topology (left, cones shown as colored balls) onto a spherical orbifold with 3 cones (computed embedding shown in the middle); this mapping is a seamless harmonic mapping to the target spherical orbifold, as can be seen by the seamless tiling of the sphere (right).

Theoretically, we prove two results: first, a generalization of the classic Tutte embedding [Floater 2003; Tutte 1963] to the spherical case (embedding meshes to a spherical convex polygonal domain). Second, given two extra, mild assumptions, we prove a Tutte embedding theorem for all spherical orbifolds. This paper also complements recent generalizations of Tutte's algorithm to orbifolds in different geometries [Aigerman and Lipman 2015, 2016]. However, the non-convexity and compactness of the spherical case necessitate different algorithmic and theoretical tools than the previous cases.

We demonstrate that our algorithm can efficiently map large models (up to a few million triangles), while in practice producing valid spherical embeddings that approximate conformal maps. We show through experiments that the positive-curvature spherical orbifolds are natural candidates for mapping surfaces with sphere-like topology (whose average Gaussian curvature is 4π), often allowing lower isometric-distortion embeddings compared to their Euclidean counterparts.

2 PREVIOUS WORK

Spherical parameterization. Parameterization to different domains is one of the most basic and important tasks in Geometry Processing and Computer Graphics; detailed surveys can be found in [Hormann et al. 2007; Sheffer et al. 2006]. Embedding a mesh into the unit sphere has long been studied. [Alexa 1999] computes a Tutte-like spring embedding, letting the edges relax while penalizing long edges so as to avoid reaching the zero map. [Haker et al. 2000] use a streographic projection to compute a conformal map to the sphere. [Shapiro and Tal 1998] extract a base tetrahedron and reinsert the vertices while preserving validity of the embedding; [Asirvatham et al. 2005; Praun and Hoppe 2003] extend this idea using progressive meshes. [Gotsman et al. 2003] build upon theory from [Lovász and Schrijver 1999] to extend the concept of barycentric coordinates to the spherical case and characterize the set of valid embeddings; [Saba et al. 2005] continue this work and suggest an efficient method to solve these equations. Computationally, they solve a system of quadratic equality constraints for which there is no guaranteed way to find a valid solution, even if one exists. [Gu et al. 2003] compute a conformal map of a spherical mesh to a sphere. [Sheffer et al. 2004] derive necessary and sufficient geometric conditions for an assignment of spherical angles to generate a valid spherical embedding. They minimize the l_2 error of these conditions, leading to a non-convex problem which is not guaranteed to reach a valid solution. [Friedel et al. 2007] also embed a mesh to the sphere by minimizing the Dirichlet energy; they minimize the energy of the graph extrinsically, while cleverly reweighing the per-triangle energy so as to ensure the modified reweighed energy gives an upper bound to the real, intrinsic energy. Flow methods such as [Kazhdan et al. 2012] and [Crane et al. 2013] can be used to smooth the mesh until it converges to a sphere. These flow methods are fast and robust, however they generate parameterizations into the entire sphere, and there is no guarantee as to whether they produce valid embeddings. [Wang et al. 2014] compute spherical parameterizations in an as-rigid-as-possible manner. [Buss and Fillmore 2001] compute averages on the sphere.

Tutte's embedding. Tutte's embedding is a method originally devised for computing injective embeddings into the Euclidean plane, $\mathbb{R}^2$; the classic Tutte embedding [Floater 2003; Tutte 1963] yields globally injective embeddings by computing a convex-combination map into a convex polygon, and has seen many uses, *e.g.*, for surface mapping [Weber and Zorin 2014]. [Gortler et al. 2006; Lovász 2004] extend Tutte's embedding to the Euclidean flat-torus case by integrating harmonic one-forms on the torus. [Aigerman and Lipman 2015] extend Tutte to the Euclidean orbifolds, *i.e.*, cone manifolds which can tile $\mathbb{R}^2$. Following the work of [Tsui et al. 2013], [Aigerman and Lipman 2016] extend Tutte's embedding beyond Euclidean domains, to hyperbolic orbifolds – cone manifolds which tile $\mathbb{H}^2$. Similarly, this work extends Tutte's embedding to spherical orbifolds - cone manifolds which tile the unit sphere.

Cone parameterization. Cone parameterization has been gaining popularity as a method to seamlessly embed closed surfaces into planar domains. The basic idea is to achieve this seamlessness by setting periodic boundary conditions [Ray et al. 2006]. The addition of cones to the parameterization is an effective way to reduce the distortion of the embedding [Myles and Zorin 2012, 2013]. Due to their seamlessness and low distortion, cone parameterizations are highly popular for the task of quad-meshing. This concept was addressed from several different points of view, such as integrating one-forms with singularities [Tong et al. 2006], producing a branched covering of the surface [Kälberer et al. 2007], modifying the metric of the surface [Chien et al. 2016], or directly computing the parameterization while enforcing the appropriate boundary conditions [Bommes et al. 2009].

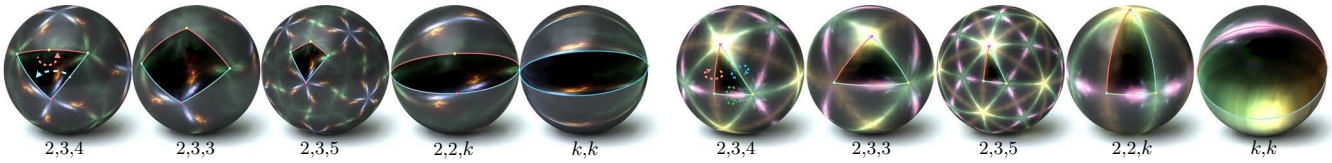


Fig. 2. The different orbifolds used in this paper, visualized by tiling the sphere with copies of the representative basic tile (in bold color). On the left are the 5 orbifolds with rotational symmetry and sphere-like topology, on the right, the 5 orbifolds with reflectional symmetry and disk-like topology.

3 PRELIMINARIES

Spherical geometry. The basic domain considered in this paper is the 2-sphere embedded in $\mathbb{R}^3$, $S^2 = \left\{x \in \mathbb{R}^3 \mid \|x\| = 1\right\}$. The *geodesic distance* $d(p, q)$ between two points $p, q \in S^2$, is the length of the shorter great arc (a part of a great circle) connecting the two points. There are several closed-form formulas for the distance; throughout this paper we shall use a specific choice:

$$d(p, q) = \operatorname{atan2}\left(\|p \times q\|, \langle p, q\rangle\right), \tag{1}$$

which was chosen specifically as it enables simplifying the optimization problem we will reach. We will also make use of *isometric transformations* of the sphere – these are the transformations that preserve geodesic distances between all points on the sphere. The group of isometric transformations of the sphere is the orthogonal group $O(3)$, comprised of all three-dimensional rotations and reflections.

Spherical orbifolds. Formally, a spherical orbifold is generated by taking the quotient of the sphere by a discrete symmetry group $G \subset O(3)$ of the sphere: given such a symmetry group G, the orbifold, denoted as $\mathcal{O}$, is defined as the set of equivalence classes of G, called *orbits*. That is, $\mathcal{O} = \left\{[p] \mid p \in S^2\right\}$, where $[p] = \{g(p) | g \in G\}$ is the orbit of $p \in S^2$. The topology of $\mathcal{O}$ is the quotient topology. Picking a representative from each orbit in $\mathcal{O}$ produces a *tile*; a tile is a subset of the sphere that can be used to tile the sphere using the transformations in the group G.

We use the term *basic tile* for a particular choice of a tile of the orbifold, namely a simple polygon on the sphere. The vertices of this polygon are symmetry points of G, *i.e.*, either gyration points (*i.e.*, rotation centers) or kaleidoscopic points (*i.e.*, meeting points of two or more reflection lines), see [Conway et al. 2008]. Each edge of the tile has a transformation from the group G associated to it, either a rotation of the edge around a symmetry point, or a reflection along the edge. Figure 2 depicts the basic tiles of 10 spherical orbifolds - 5 with rotational symmetry (left) and 5 with reflectional symmetry (right). For each of the two symmetry types, we show one example with arrows that visualize the symmetry transformations associated to different sides of the tile; these transformations can be thought of as the "assembly instructions" guiding us how to move and connect copies of the tile so as to tile the entire sphere, by applying these transformations to the basic tile repeatedly. The tile then represents the orbifold, by considering a point p of the tile and identifying all its generated copies $[p]$ in the tiling as being the same point.

The spherical orbifolds are a subfamily of the constant positive curvature cone-surfaces. They have either sphere-like or disk-like topology, and cones of integer order (*i.e.*, cone angles of $2\pi/n, n \in \mathbb{N}$). Our goal is to derive a practical method for computing Tutte embeddings into spherical orbifolds, and prove they are valid (bijective) embeddings.

4 APPROACH

In this section we review the full details of our method. We first define the spherical orbifold Tutte embeddings, and then devise an algorithm for their computation.

Spherical embeddings. Given a 3-connected polygonal mesh $\mathbf{M}$ with vertices $\mathbf{V} = \{v_i\}$, edges $\mathbf{E} = \{(i, j)\}$ and faces $\mathbf{T}$, we define the image of each vertex v_i to to be a variable $\Phi_i \in S^2 \subset \mathbb{R}^3$. These are the degrees of freedom of the map, which are concatenated into a 3-by-$|\mathbf{V}|$ matrix representing the full map, $\Phi = \left[\Phi_1, \Phi_2, \ldots, \Phi_{|\mathbf{V}|}\right]$. The image of each edge $(i, j) \in \mathbf{E}$ is set to be the shortest geodesic connecting the images of the two vertices Φ_i, Φ_j (in case the points are antipodal one of the geodesics can be chosen arbitrarily). Each triangle of the original mesh is thus mapped to a spherical triangle, and the map can be extended to any point on the surface defined by $\mathbf{M}$, *i.e.*, also to points on an edge or inside a triangle, using spherical barycentric coordinates.

Harmonic spherical maps. The spherical Dirichlet energy of an embedding Φ is defined as a weighted sum of squared spherical distances between all adjacent vertices:

$$E(\Phi) = \sum_{(i,j)\in\mathbf{E}} w_{ij}\, d(\Phi_i, \Phi_j)^2, \tag{2}$$

for some set of fixed positive weights $w_{ij} > 0$. Although any set of positive weights is valid for our method, throughout this work we will use the cotangent weights [Pinkall and Polthier 1993]. In case any of the weights are negative we clamp them to 0.01 to ensure they are all positive.

We will compute *discrete harmonic* embeddings, which are defined as critical points of E. Discrete harmonic embeddings have been used extensively in the past [Floater 2003], and also in the spherical case [Friedel et al. 2007; Gotsman et al. 2003; Saba et al. 2005]. In this sense, the main novelty of our method is in the boundary conditions we set, which define a mapping to a spherical cone surface, and are the reason the embeddings are valid.

Embedding into an orbifold. Our goal is to devise a method to embed the given mesh into a spherical orbifold $\mathcal{O}$. For simplicity of the following explanation, let us for now focus on a specific scenario, as depicted in Figure 3: the given input mesh $\mathbf{M}$ has sphere-like topology, and the target orbifold has 3 cones. The other cases will be covered later, as simple extensions to the method presented next. To compute the orbifold embedding we follow three steps:

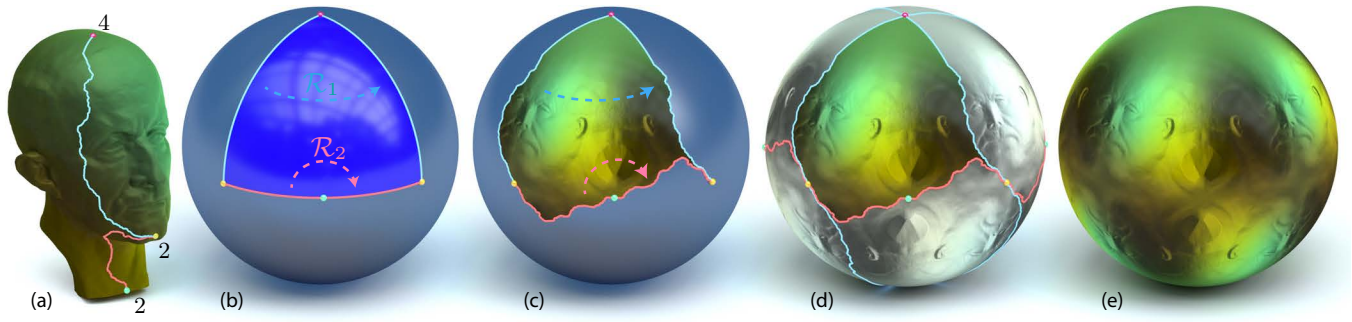


Fig. 3. The outline of our method. **(a)** the input to our algorithm consists of an input mesh and a discrete set of cones (in this case there are 3, visualized as colored balls) and an integer n_i associated to each cone denoting its symmetry-order. **(b)** according to the number of cones and their order, we construct a basic tile describing the boundary conditions of the embedding; in this specific tile, the two blue edges are associated by a 90° rotation, $\mathcal{R}_1$, and the two red edges are associated by a 180° rotation, $\mathcal{R}_2$. **(c)** the output of our algorithm – an embedding of the cut mesh into the sphere, which minimizes the spherical Dirichlet energy under the boundary conditions defined according to the basic tile. **(d)** the specific boundary conditions ensure that we can tile the entire sphere with rotated copies of the embedding – this is the special symmetry property of an orbifold embedding. The resulting tiled embedding is a bijective, discrete harmonic and (as shown in **(e)**) seamless embedding into the orbifold. The number of rotated copies of the mesh around each cone respects the symmetry order associated to the cone.

(1) Choose the 3 vertices of the mesh that will be mapped to the cones of the orbifold, and assign a symmetry order to each cone, according to one of the possible cone symmetry order assignments.
(2) Cut the mesh through the cones to produce a disk-like mesh and construct the boundary conditions generating the orbifold structure.
(3) Compute a critical point of the Dirichlet energy, subject to the boundary conditions.

Step 1 relies on user-input. We detail steps 2-3 next.

4.1 Cutting and boundary conditions

At this stage we are given the mesh $\mathbf{M}$ with sphere-like topology, the choice of the 3 vertices $c_1, c_2, c_3 \in \mathbf{V}$ which will be mapped to the cones, and the target symmetry order of each cone c_i, denoted as n_i (see Figure 3 (a)). Our goal is to cut the mesh $\mathbf{M}$ open to a topological disk, and assign the relevant boundary conditions to the boundary generated by the cutting.

Cutting the mesh. We cut $\mathbf{M}$ to a topological disk by cutting through the cones, as shown in Figure 4 (the color scheme for the cuts and cones is consistent with Figure 3): as depicted on the left, we trace a path of edges, denoted γ_1, connecting the first two cones c_1, c_2 (*e.g.*, using Dijkstra's shortest path algorithm).

We repeat this procedure for c_2, c_3 to create the path γ_2. We now cut the mesh through the two paths as depicted on Figure 4, right: we traverse the edges on the cut, for each edge find the two triangles adjacent to it, and disconnect them from one another. In the process of separating the triangles, each vertex v_i on the cut is duplicated into two vertices on the two sides of the cut, which we denote as $v_i^{\triangleleft}, v_i^{\triangleright}$. Hence, the path γ_1 is duplicated into two corresponding paths, $\gamma_1^{\triangleleft}$ and $\gamma_1^{\triangleright}$. Likewise, γ_2 is duplicated into $\gamma_2^{\triangleleft}, \gamma_2^{\triangleright}$. Lastly, note that the cones c_1, c_3 are not duplicated, since they are the end-points of the cut; on the other hand the middle cone c_2 is duplicated and will appear two times in the embedding as $c_2^{\triangleleft}, c_2^{\triangleright}$, so after cutting

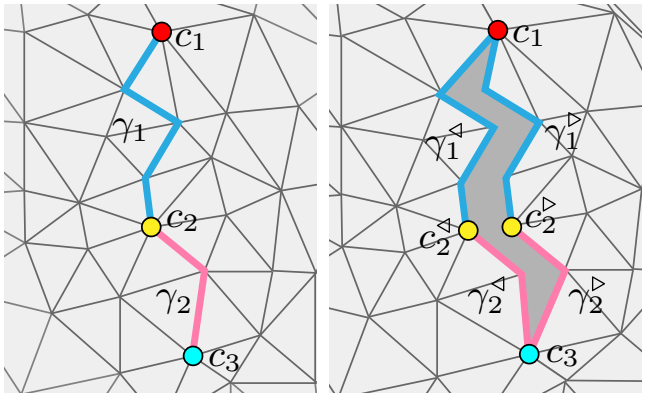


Fig. 4. The cutting scheme used in this paper. Left: the 3 vertices selected as cones are connected with two edge-paths; right: cutting the mesh through the paths duplicates vertices and edges along the cut into corresponding copies.

the 3 original cones are now represented as 4 boundary vertices. Henceforth we shall no longer need to discuss the original input mesh, and therefore henceforth abuse notation and also denote by $\mathbf{M}$ the *cut* mesh produced from the input mesh. We denote by $\partial\mathbf{M}$ the boundary of the cut mesh, which is comprised of $\gamma_1^{\triangleleft}, \gamma_1^{\triangleright}, \gamma_2^{\triangleleft}, \gamma_2^{\triangleright}$. We denote by $C = \{c_1, c_2^{\triangleleft}, c_2^{\triangleright}, c_3\}$ the set of 4 boundary vertices corresponding to the cones.

Boundary conditions. We now have at our disposal the cut mesh $\mathbf{M}$ with disk-like topology, its boundary $\partial\mathbf{M}$ segmented into two pairs of corresponding paths $\gamma_1^{\triangleleft} \leftrightarrow \gamma_1^{\triangleright}, \gamma_2^{\triangleleft} \leftrightarrow \gamma_2^{\triangleright}$. Now, according to the assigned symmetry order of each of the three cones, n_1, n_2, n_3, we compute the suitable basic tile structure. The basic tile consists of assigning a position $p_i \in S^2$ to each one of the four cone-points $c_1, c_2^{\triangleleft}, c_2^{\triangleright}, c_3$, and a rotation $\mathcal{R}_i$ to each pair of corresponding paths

$\gamma_i^{\triangleleft}, \gamma_i^{\triangleright}$. The basic tile, corresponding to the cut mesh in Figure 4, is shown in Figure 3 (b) (with the same coloring scheme of boundary segments and cones). The positions $\{p_i\}$ and rotations$\{\mathcal{R}_i\}$ of the basic tile are decided solely according to the topology of the mesh, the number of cones, and the cones' symmetry order; the construction process is detailed in Appendix A. This provides us with the boundary conditions of the orbifold embedding: we require for each pair of corresponding paths, that their images are identical up to the relevant transformation $\mathcal{R}_i \in O(3)$ of that segment of the boundary, *i.e.*,

$$\gamma_i^{\triangleleft} = \mathcal{R}_i\left(\gamma_i^{\triangleright}\right), \quad i = 1, 2$$

where we slightly abused notation and denoted by γ_i the *image* of the boundary path on the sphere. The transformations $\mathcal{R}_1, \mathcal{R}_2$ are illustrated with arrows in Figure 3.

This set of boundary conditions is needed to ensure the result will be a tile (according to the definition presented in Section 3) and therefore that the embedding well-defines a map into the relevant orbifold. To satisfy the boundary conditions, we consider corresponding vertices $v_i^{\triangleleft}, v_i^{\triangleright}$ on the two sides of the cut, and require that their images are identical up to the corresponding transformation R_i:

$$\Phi_i^{\triangleleft} = R_i \Phi_i^{\triangleright}, \quad R_i = \begin{cases} \mathcal{R}_1 & v_i \in \gamma_1 \\ \mathcal{R}_2 & v_i \in \gamma_2 \end{cases}. \tag{3}$$

Lastly, we fix the 4 cone vertices $c_1, c_2^{\triangleleft}, c_2^{\triangleright}, c_3$ to their assigned positions,

$$\Phi_i = p_i \quad , v_i \in C. \tag{4}$$

4.2 Computing the embedding

Given the mesh $\mathbf{M}$ with the boundary conditions (3) and (4), we can now compute the orbifold Tutte embedding by computing a critical point of the spherical Dirichlet energy under the orbifold boundary conditions, *i.e.*, find a solution to the following optimization problem:

$$\begin{aligned} \min \quad & E(\Phi) && && (5a)\\ \text{s.t.} \quad & \|\Phi_i\| = 1 && \forall v_i \in \mathbf{V} && (5b)\\ & \Phi_i^{\triangleleft} = R_i \Phi_i^{\triangleright} && \forall v_i \in \partial\mathbf{M} && (5c)\\ & \Phi_i = p_i && \forall v_i \in C && (5d) \end{aligned}$$

where $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^3$.

Intuitively, Equation (5) seeks a discrete harmonic map into a tile of the target spherical orbifold. This embedding indeed defines a map into the orbifold, as each vertex v_i is mapped to a well-defined point on the orbifold, namely the orbit $[\Phi_i]$. Note that $\left[\Phi_i^{\triangleleft}\right] = \left[\Phi_i^{\triangleright}\right]$ and hence the twin vertices $v_i^{\triangleleft}, v_i^{\triangleright}$ are mapped to the same orbifold point, thus the original vertex v_i is well-defined to be mapped to $[\Phi_i]$, and the embedding indeed represents a homeomorphism between the *uncut* mesh to the orbifold O.

Problem (5) is challenging mainly due to the non-convex constraint (5b); next we elaborate on how to practically solve this problem via a reduction to a simpler optimization problem.

Equivalent formulation. We compute the embedding Φ by finding a critical point of problem (5). To that end, we first show that relaxing the problem by simply removing the non-convex unit norm constraint (5b) leads to an *equivalent* formulation: we remove (5b), now letting Φ_i be any point in the ambient space $\mathbb{R}^3$; we treat Φ_i as representing the projected point on the sphere, $\Pi(\Phi_i) \triangleq \Phi_i / \|\Phi_i\|$. For this projection to be well-defined we require $\Phi_i \neq 0$. We now show this leads to an equivalent problem to problem (5).

First, we note that the distance function (1) is invariant to scaling of its arguments, that is, for $\lambda_1, \lambda_2 > 0$

$$d(\lambda_1 p, \lambda_2 q) = d(p, q)$$

and therefore also the spherical Dirichlet energy (2) satisfies

$$E\left(\Phi \cdot \mathrm{diag}(\lambda_1, \ldots, \lambda_{|\mathbf{V}|})\right) = E(\Phi),$$

i.e., scaling the image Φ_i of each vertex by an arbitrary positive scalar does not change the energy, in particular $E(\Phi) = E(\Pi(\Phi))$.

As for the boundary conditions, we note that if $\Phi_i \neq 0$ satisfies the linear constraint (5c), its projection onto the sphere also satisfies the same constraint, *i.e.*, if $\Phi_i^{\triangleleft} = R_i \Phi_i^{\triangleright}$, then

$$\Pi(\Phi_i^{\triangleleft}) = \frac{\Phi_i^{\triangleleft}}{\left\|\Phi_i^{\triangleleft}\right\|} \overset{(5c)}{=} \frac{R_i \Phi_i^{\triangleright}}{\left\|R_i \Phi_i^{\triangleright}\right\|} \overset{R_i \in O(3)}{=} R_i \frac{\Phi_i^{\triangleright}}{\left\|\Phi_i^{\triangleright}\right\|} = R_i \Pi(\Phi_i^{\triangleright}).$$

Thus, an equivalent problem to (5) is:

$$\begin{aligned} \min \quad & E(\Phi) && && (6a)\\ \text{s.t.} \quad & \Phi_i \neq 0 && \forall v_i \in \mathbf{V} && (6b)\\ & \Phi_i^{\triangleleft} = R_i \Phi_i^{\triangleright} && \forall v_i \in \partial\mathbf{M} && (6c)\\ & \Phi_i = p_i && \forall v_i \in C && (6d) \end{aligned}$$

which has the same set of constraints as (5), except for (5b) which is replaced with (6b). Practically, this constraint can be enforced by a simple scaling of the variables during optimization, when any vertex is detected to get close to the origin during the line search. In practice, this process was never applied as vertices never strayed close to the origin; the only way this may happen is if the line-search direction is proportional to Φ_i, which entails the unlikely scenario of the line-search being *orthogonal* to the gradient $\nabla_{\Phi_i} E$.

Optimization . The main reason to cast the original problem (5) we aim to solve, into problem (6), is that it leads to a simple optimization problem consisting of minimizing a non-convex, smooth functional, subject to linear equality constraints. Motivated by [Kovalsky et al. 2016; Liu et al. 2016] we solve this optimization problem using a Laplacian-preconditioned L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) algorithm. We initialize the optimization with a feasible starting point by projecting the Euclidean Tutte embedding back onto the sphere, as illustrated in Figure 13, bottom. We elaborate on all details in appendix B.

4.3 The other spherical orbifolds

Up until now, for the sake of simplicity, we have considered only cases in which the orbifold is of sphere-like topology and has 3 cones. We now discuss the other cases, which also lead to boundary conditions formulated exactly as (3).

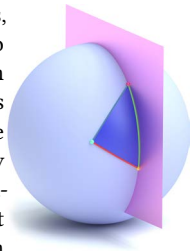
The variety of spherical orbifolds is finite: there are exactly 14 types of spherical orbifolds ([Conway et al. 2008], Chapter 4), which we enumerate next.

Orbifolds with reflective symmetry points. These orbifolds are depicted in Figure 2, right. They all correspond to *disk-like* topology, and are generated from a group of reflections (visualized as arrows on the 5$^{\text{th}}$-from-right sphere in Figure 2), as opposed to the orbifolds we considered up until now, which had sphere-like topology and were generated from a group of rotations. In the reflective orbifolds, each side of the boundary of the basic tile has a reflection associated to it. The points where two or more reflection lines meet are called *reflective cones*, and their order counts how many reflection lines cross that point.

In practice, this means that the selected vertices which are mapped to the reflective cones lie on the boundary of the mesh, segmenting the boundary into parts. We order the cones in clockwise order. We connect each two consecutive cones c_i, c_{i+1} with a path passing on the boundary of the mesh, denoted as before as γ_i. We now require that the image of each path is identical to itself under reflection of the corresponding edge of the basic tile (see Appendix A), that is, $\gamma_i = \mathcal{R}_i(\gamma_i)$, or in terms of the vertices,

$$\Phi_i = R_i \Phi_i, \tag{7}$$

where as before, $R_i = \mathcal{R}_j$ for j which satisfies $v_i \in \gamma_j$. To be consistent with the previous case, we can simply use the notation $\Phi_i^{\blacktriangleleft}$ and $\Phi_i^{\blacktriangleright}$. However here we do not cut and duplicate vertices, but instead only use this notation in order to cast Equation (7) above into the format of Equation (3), by writing $\Phi_i^{\blacktriangleleft} = \Phi_i^{\blacktriangleright} = \Phi_i$. This gives a set of boundary conditions formulated just as in the previous case.



Note that practically, as the inset illustrates, as Eq. (7) requires that a point is mapped to itself under a given reflection (*e.g.*, points on the green segment of the boundary), this entails the point lies *on* the reflective plane (in purple in the inset). So, geometrically, these boundary conditions require that all vertices on a boundary segment γ_i are restricted to lie on the great circle supporting the edge of the basic polygon corresponding to γ_i (however they may lie anywhere on the great circle and not necessarily on the interior of the basic tile, *i.e.*, are free to "slide" and are not fixed to place).

2-cone orbifolds. There are a few orbifolds with only two cones, at the north and south poles. To accommodate for this case, the previous construction needs only minor modifications: in case the mesh if sphere-like, it will only have a single cut γ_1 passing between the two cone vertices, and the two sides of the cut in the embedding are related by a rotation around the z-axis by the cone angle $\alpha = 2\pi/k$ where the symmetry order k can be any positive integer. In the reflective orbifold case, the boundary will be segmented only to two sides rather than three.

Lastly, there are 4 more orbifolds which have a mixture of rotational and reflective points; they are either topologically disk-like, or projective-plane-like.

5 PROPERTIES

In this section we prove some properties of embeddings defined as critical points of spherical Dirichlet problems such as (6). We will prove two results: first, a spherical version of the classic Tutte algorithm [Floater 2003; Tutte 1963], regarding bijective embeddings to convex spherical polygonal regions; second, we show that under two extra assumptions, Φ which is a critical point of (6) produces a bijective embedding to the respective orbifold.

We start with proving the validity of the spherical Tutte embedding for disk-like surfaces, which can be seen as a generalization of the classic Tutte algorithm to spheres. Although this result seems fundamental, we haven't encountered it in previous works.

Theorem 1. *Let $\boldsymbol{M} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{T})$ be a disk-like 3-connected triangular mesh, and consider the spherical Dirichlet problem:*

$$\begin{aligned} \min \quad & E(\Phi) && && (8a)\\ \text{s.t.} \quad & \Phi_i \neq 0 && \forall v_i \in \boldsymbol{V} && (8b)\\ & \Phi_i = p_i && \forall v_i \in \partial \boldsymbol{M} && (8c) \end{aligned}$$

where $p_i \in S^2$ are vertices of a convex polygon Ω on the sphere, and the assignment $\Phi_i \leftrightarrow p_i$ defines a homeomorphic boundary map $\partial \boldsymbol{M} \to \partial\Omega$. Let H be an arbitrary hemisphere containing Ω. Then, (i) there exists a critical point of (8) contained in H; and (ii) every critical point of (8) contained in H defines a bijection between $\boldsymbol{M}$ and Ω.

The proof is based on a reduction to the classic planar case via the *gnomonic* projection. Note that a convex spherical polygon is contained inside a closed hemisphere (this hemisphere may not be unique); denote such a hemisphere as H. The proof follows the convex combination maps theorem [Floater 2003] using the following three lemmas:

Lemma 1. *Problem (8) has a critical point Φ inside H.*

The lemma is proved in Appendix C. Henceforth we consider an arbitrary critical point $X \subset H$.

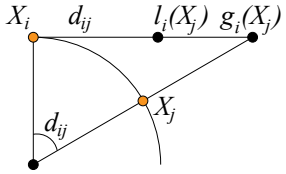


Fig. 5. Inverse exponential and gnomonic projections.

The second lemma considers the gnomonic projection with respect to a hemisphere H. The gnomonic projection of $X_j \in S^2$ w.r.t. the hemisphere H centered at X_i is defined as $g_i(X_j) = X_j / \langle X_j, X_i \rangle$, projecting points to the plane defined via the equation $\{x \in \mathbb{R}^3 | \langle x, X_i \rangle = 1\}$; see Figure 5 for an illustration.

Lemma 2. *The gnomonic projection maps spherical convex polygons contained in H to Euclidean convex polygons.*

The lemma is proved in Appendix C.

The third lemma is proved in [Buss and Fillmore 2001] (see Theorem 3),

Lemma 3. *A critical point X_i of the spherical averaging energy $e_i(\cdot) = \sum_{j \in \mathcal{N}_i} w_{ij} d(\cdot, X_j)^2$ satisfies*

$$\sum_{j \in \mathcal{N}_i} w_{ij} \left(\ell_i(X_j) - X_i \right) = 0, \tag{9}$$

where $\ell_i(X_j)$ is the inverse exponential map centered around X_i, applied to X_j, and $\mathcal{N}_i$ is the set of indices of neighbors of the i-th vertex.

See Figure 5 for an illustration of the inverse exponential map $\ell_i(X_j)$. Now let Φ be a critical point of (8). The following relation

between the inverse exponential map and the gnomonic map holds:

$$g_i(X_j) - X_i = \frac{\tan d_{ij}}{d_{ij}} \left(\ell_i(X_j) - X_i\right). \tag{10}$$

where $d_{ij} = d(\Phi_i, \Phi_j)$. Eq. (10) combined with (9) implies that $g_i(X_i)$ is a Euclidean convex combination of its neighbours' images, $\{g_i(X_j)\}_{j\in\mathcal{N}_i}$. Let g_0 be the gnomonic projection w.r.t. the centroid of H. Consider the composition $\Psi = g_0 \circ \Phi$, where Φ is the map induced by the positions X. Ψ maps $\partial\mathbf{M}$ to a boundary of a convex Euclidean polygon, due to Lemma 2. Now, for any i, $g_0 \circ g_i^{-1}$ is a perspective transformation and therefore preserves convex combinations. Hence $\Psi = g_0 \circ g_i^{-1} \circ g_i \circ \Phi$ is a convex-combination map, and therefore, due to the convex-combination map theorem by Floater [Floater 2003], Ψ is bijective, entailing $\Phi = g_0^{-1} \circ \Psi$ is bijective. This proves Theorem 1.

To address the general spherical orbifold case we first use the symmetry group G to tile the mesh $\Phi(\mathbf{M})$ over the sphere, *e.g.*, Figure 3 (d) and (e). We denote this new spherical mesh $\mathbf{M'} = (\mathbf{V'}, \mathbf{E'}, \mathbf{T'})$. $\mathbf{M'}$ is made up of rotated and reflected copies of $\Phi(\mathbf{M})$, hence if we show that $\mathbf{M'}$ is a valid spherical triangulation, it will imply that Φ is a bijective map from the uncut $\mathbf{M}$ to the orbifold O.

We will follow [Gotsman et al. 2003; Saba et al. 2005] and use the spherical embedding theory in [Lovász and Schrijver 1999]. As discussed in these works, a sufficient condition for a 3-connected $\mathbf{M'}$ to be a valid spherical triangulation is that its coordinate vectors $X = \left[X_1, X_2, \ldots, X_{|\mathbf{V'}|}\right] \in \mathbb{R}^{3\times|\mathbf{V'}|}$ are in the kernel of a *CdV matrix* W, that is $WX^T = 0$. A matrix $W \in \mathbb{R}^{|\mathbf{V'}|\times|\mathbf{V'}|}$ is a CdV matrix if it satisfies the following two conditions: (i) for every $i \neq j$, if $(i,j) \in \mathbf{E'}$, $W_{ij} < 0$, otherwise $W_{ij} = 0$; (ii) W has exactly one negative eigenvector. Two comments are in order: first, the Perron-Frobenious theorem can be used to show that any matrix satisfying condition (i) has at-least one negative eigenvector (condition (ii) requires there to be exactly one). Second, CdV matrices have some similarities to Laplacian matrices with the difference that their diagonal is arbitrary.

We will next show that there exists a matrix W satisfying condition (i) and that the coordinate vectors X are in its kernel. We make the assumption that the maximal spherical edge lengths in $\mathbf{W'}$ is smaller than $\pi/2$, that is $d_{ij} < \pi/2$, for all $(i,j) \in \mathbf{E'}$. We define W to be the matrix corresponding to the following linear system:

$$X_i \left(\sum_{j\in\mathcal{N}_i} \gamma_{ij} \right) - \sum_{j\in\mathcal{N}_i} \frac{\gamma_{ij}}{\cos(d_{ij})} X_j = 0, \quad i = 1, \ldots, |\mathbf{V'}|, \tag{11}$$

where $\gamma_{ij} = w_{ij} \frac{d_{ij}}{\tan(d_{ij})}$.

First, we note that since $d_{ij} = d_{ji}$ (the distance is symmetric), W is also symmetric and satisfies condition (i) of the CdV matrices. To show that $WX^T = 0$ we first note that each vertex position X_i in $\mathbf{M'}$ is a critical point of the spherical average energy $e_i(\cdot) = \sum_{j\in\mathcal{N}_i} w_{ij} d(\cdot, X_j)^2$, where w_{ij} are the weights defined on the edges of the original $\mathbf{M}$. This is true also at the cone vertices of $\mathbf{M}$ due to the symmetry of the corresponding one-ring at $\mathbf{M'}$. This means we can use Lemma 3 on X. Eq. (11) can now be shown by combining equations (9), (10) and $\left\|g(X_j)\right\| = \frac{1}{\cos(d_{ij})}$. See Figure 5 again for an illustration. We proved:

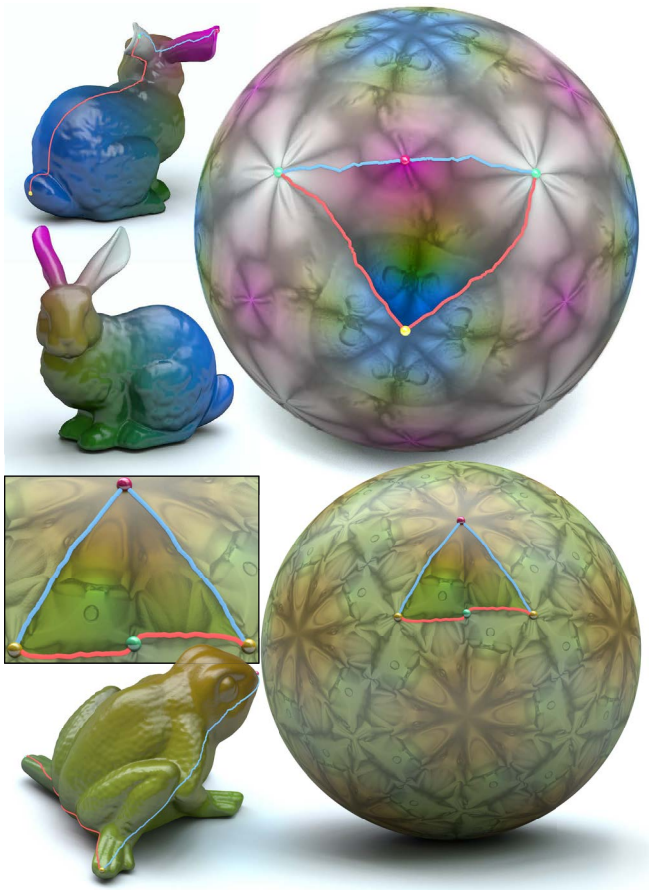


Fig. 6. Embeddings of sphere-like meshes to sphere-like orbifolds with cones of order 4, 3, 2 (top) and 5, 3, 2 (bottom).

Theorem 2. *Let $\boldsymbol{M} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{T})$ be a disk-like 3-connected triangular mesh, and Φ a critical point of* (6)*. Let $\boldsymbol{M'}$ be the cover spherical triangulation, and W the matrix defined via* (11)*. If: (i) the maximal edge length in $\boldsymbol{M'}$ is smaller than $\pi/2$; and (ii) W has a single negative eigenvector, then $\boldsymbol{M'}$ is a valid spherical triangulation and Φ defines a bijection from the original* uncut *mesh $\boldsymbol{M}$ into the spherical orbifold O.*

Theorem 2 makes two assumption regarding the embedding Φ. The 2nd assumption excludes critical points of the Dirichlet energy that are in a non-trivial homotopy class (*e.g.*, a double-wrapping of the sphere). Although we could not prove these assumptions always hold, and in general, numerical descent methods are not guaranteed to preserve the homotopy class of the map, we conjecture these additional assumptions are always satisfied by the output of our algorithm, based on the numerical experiments performed in this paper.

6 EXPERIMENTS

We have tested our algorithm on a wide variety of meshes. In all cases our algorithm produced a bijective, seamless embedding, approximating a conformal map into the desired orbifold. We visualize each embedding by drawing it on the sphere in bold color, and

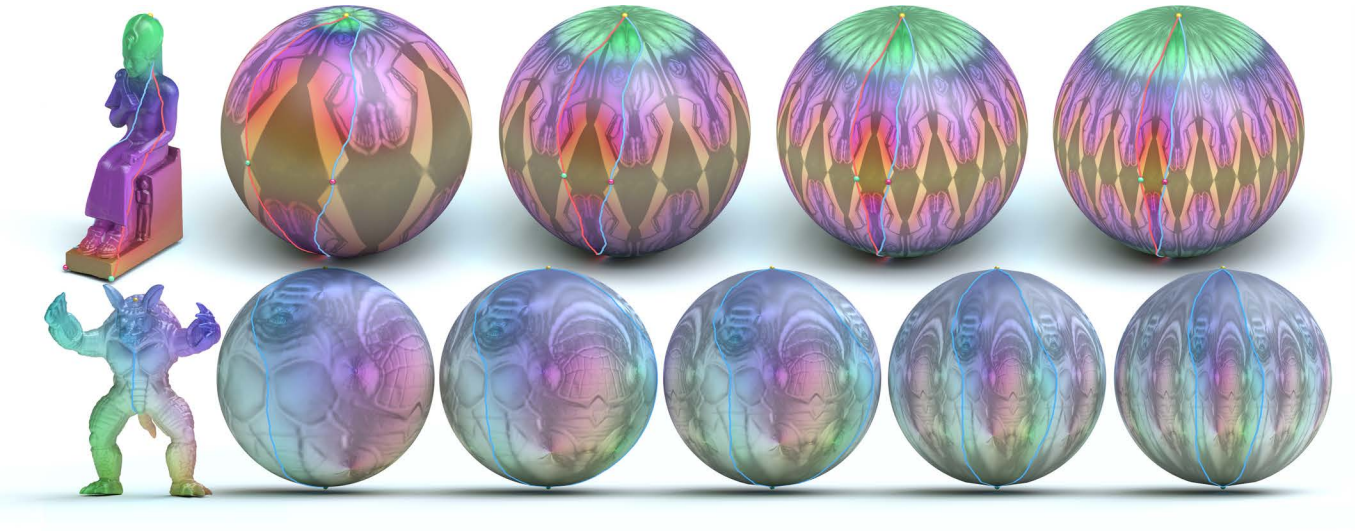


Fig. 7. Top: an embedding of a sphere-like mesh to a 3-cone sphere orbifold with cones of order 2, 2, k, where k can be be any integer larger than 1; we show $k = 4, 8, 12, 16$. Bottom: an embedding of a sphere-like mesh to a 2-cone sphere orbifold, where the two cones are placed at the north and south poles and have the same order k. The results shown are for $k = 2, 3, 4, 6, 8$.

tiling the sphere with copies of the embedding in lighter color. We also compare our algorithm to the *Euclidean* orbifold Tutte embeddings, and validate the properties of the embeddings, namely their seamlessness and conformality.

6.1 Sphere-topology orbifolds

There are 5 orbifolds with sphere-like topology where cones are *rotational* symmetry points, and their symmetry order n_i measures the angle of rotation around that point as $2\pi/n_i$. We enumerate these orbifolds in the following figures.

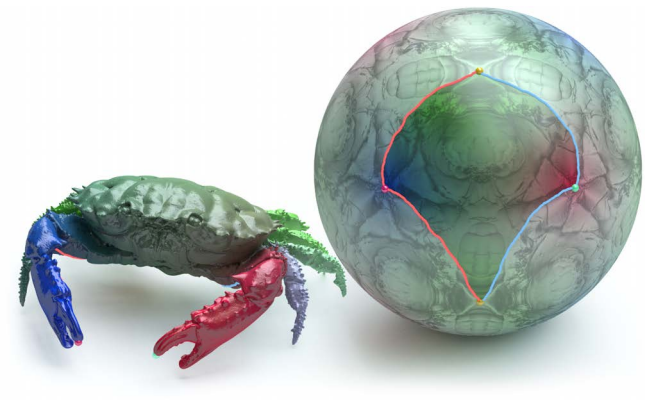


Fig. 8. An embedding of a sphere-like mesh of 2M triangles to a sphere-like orbifold with cones of order 3, 3, 2.

Figure 6 shows an embedding to an orbifold with cones of order 4, 3, 2 (top) and 5, 3, 2 (bottom). Figure 7, top, shows an embedding of a mesh to an orbifold with cones of order $2, 2, k$, where k can be any integer larger than 1; in this figure we show results for $k = 4, 8, 12, 16$. In all cases the resulting embeddings are conformal. Note how different choices for the order of the yellow cone affect the proportions of the head and feet. The bottom row shows an embedding of the armadillo to a 2-coned orbifold, where the cone order of the two cones, k, k, can again be chosen to be any integer greater than 1 – we show $k = 2, 3, 4, 6, 8$. Figure 8 shows an embedding of a mesh to an orbifold with cones of order 3, 3, 2. This mesh consists of 2 million triangles, and our optimization produced the bijective embedding in 350 seconds.

6.2 Disk-topology orbifolds



Fig. 10. An embedding of a flat disk mesh to a disk-like orbifold with reflective cones of order 4, 3, 2.

There are 5 spherical orbifolds generated by *reflective* lines, all with disk topology. In these cases the "cones" are meeting points

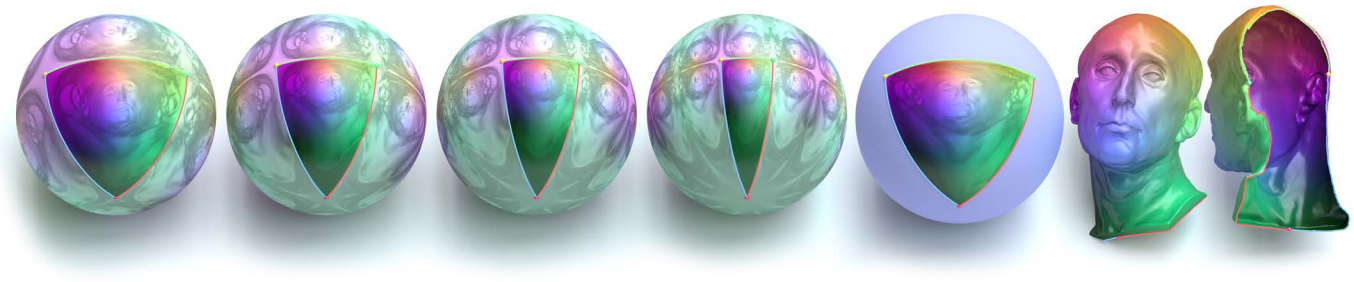


Fig. 9. From left to right: 4 embeddings to a disk-like orbifold with reflective cones of 2, 2, k, where $k = 2, 3, 4, 6$, and one "classic" Tutte embedding, in which the boundary is fixed to place – note this embedding exhibits shear and is not conformal, as opposed to the leftmost orbifold embedding which produces a conformal map into the same domain.

of the reflection-lines, and their symmetry order n_i denotes how many reflection lines meet at that cone.

Figure 9 shows an embedding into an orbifold with reflective cones of order $2, 2, k$, with $k = 2, 3, 4, 6$; all the maps are conformal. In comparison, the rightmost embedding in Figure 9 is the adaptation of the "classic" Tutte embedding to the spherical case, where the boundary of the mesh is fixed into place. This map exhibits shear, and is *not* conformal, in comparison to the leftmost orbifold embedding which is a conformal embedding into the same domain. However, as we show in the inset, the classic spherical Tutte embedding is meritable on its own, as it can be used to injectively embed a mesh into *any* spherical convex domain, also ones which do not have a corresponding orbifold.

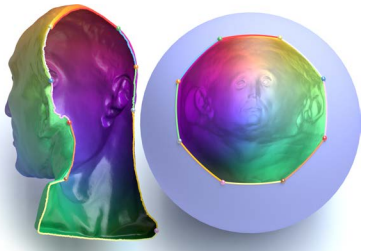


Figure 10 shows an embedding of a flat disk-like mesh to an orbifold with reflective cones of order of $4, 3, 2$. Although the circular mesh is mapped to a domain with sharp corners, the result still approximates a conformal map. In Figure 11 we show embedding into the 3 other reflective orbifolds, having cones of order $5, 3, 2$ (top) $3, 3, 2$ (middle) and k, k (bottom), in this example $k = 3$.

6.3 Comparison to Euclidean orbifold embeddings

The Euclidean counterpart of the spherical orbifold embeddings introduced in [Aigerman and Lipman 2015], provides harmonic paramterizations into the Euclidean orbifolds, which similarly to the spherical case approximate conformal maps. It has the benefit of being computationally cheaper, requiring only solving a sparse linear system.

Next we present empirical evidence that the spherical orbifolds enable conformal parameterizations with lower area distortion. This is due to two main reasons: first, the spherical orbifolds have different cone structures than the Euclidean ones, and there are orbifolds in which one can adjust the angle of the cone to be $2\pi/k$ for any integer $2 \leq k$; second, the spherical orbifolds have constant positive curvature, hence spherical surfaces which also have close-to-constant positive curvature can be embedded in the spherical orbifolds in a more isometric manner.

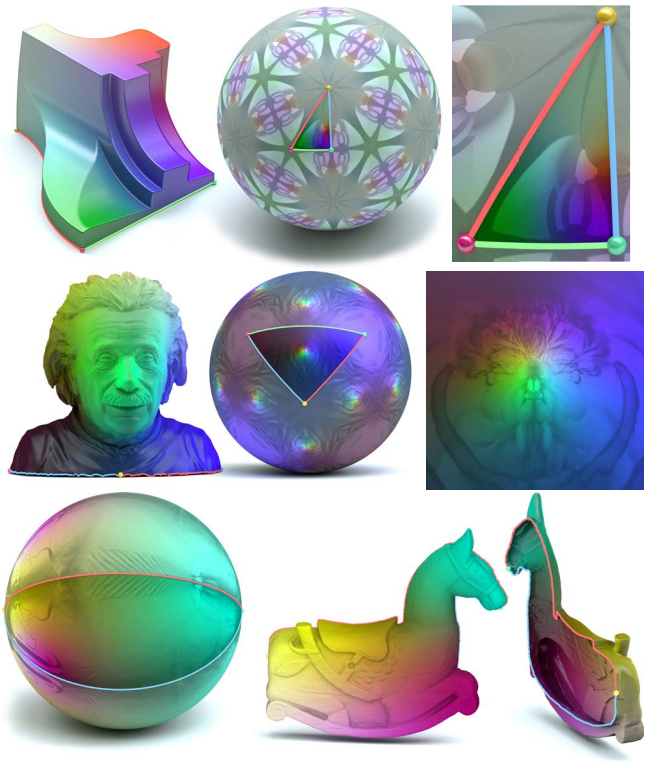


Fig. 11. Embeddings of three disk-like meshes to disk-like orbifolds with reflective cones of order 5, 3, 2 (top), 3, 3, 2 (middle), and k, k, with $k = 3$ (bottom).

In Figure 12 we show three comparisons of spherical orbifold Tutte embeddings to a Euclidean orbifold Tutte embeddings in terms of area distortion. We compute the area distortion by first rescaling the source mesh to have an area of 1, and likewise rescaling the two embeddings so that the total area of each of the two target domains is 1. We then define the distortion of a triangle $t \in \mathbf{T}$ as $d_t = \max\left(A_t, A_t^{-1}\right)$, where A_t is the change in area of t induced by the map, *i.e.*, $A_t = |\Phi(t)|\,/|t|$ (in the spherical case we compute the area using the spherical area formula). We color the meshes according to the log of the distortion, and display a histogram showing the distribution of the distortion. In the embedding of Ramesses, the

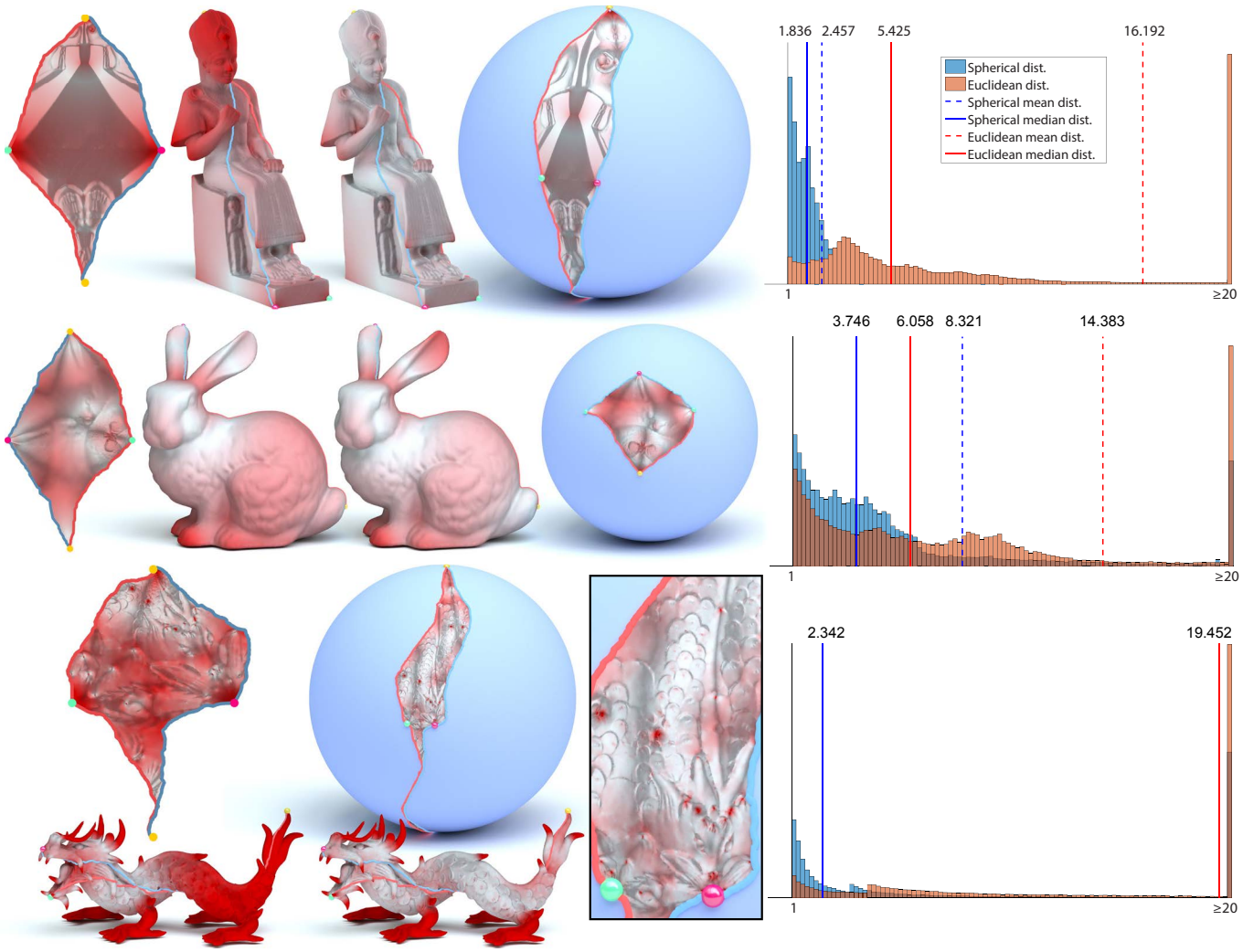


Fig. 12. Comparison of the spherical orbifold Tutte embeddings (to the right in each row) to the Euclidean orbifold Tutte embeddings (to the left in each row). Both methods produce conformal maps, however the spherical orbifolds enable generating embeddings with lower area distortion (meshes colored according to distortion). The histogram of the distortion is shown next to each example.

Euclidean orbifold embedding shrinks the head to the point that it is indiscernible, and the feet are blown up, compared to the rest of the mesh. In contrast, the spherical orbifold embedding prevents the harsh shrinking of the head, and preserves good proportions between all parts of the body. Likewise, in the embedding of the dragon, the spherical embedding embeds the body of the dragon with relatively-constant scale, as shown in the blowup, while in the Euclidean embedding the body rapidly shrinks as it approaches the tail. In this example the mean and standard deviation of the distortion were larger than 10^5 for both examples and hence are not shown in the histogram. The embeddings of the stanford bunny may seem more similar to one-another than the previous examples, as in this case the two orbifolds have a rather similar cone structure (4, 3, 2 for the spherical, 3, 3, 3 for the Euclidean), however the Euclidean orbifold has much higher average and median distortion (this can be observed as the mesh of the Euclidean embedding is slightly redder, remembering that the coloring is by log-scale).
Note that in all results, the histograms of the Euclidean embeddings have a longer “tail" of triangles with distortion larger than 20.

6.4 Properties of the embeddings

Seamlessness. The spherical orbifold Tutte embeddings are completely agnostic to the cuts, as shown in Figure 13, top two rows: we computed one embedding (top row), and then, without modifying the cones, modified the cuts on David’s head in an arbitrary manner (middle row). As expected, while each cut produces a different *tile*, they both represent exactly the same spherical orbifold embedding, as can be seen once the sphere is tiled (left column). At the bottom row, we show the initialization of our algorithm, which is also a

valid embedding into the orbifold, however it is not a spherical orbifold *Tutte* embedding, *i.e.*, not a solution to our basic optimization problem (5), and therefore is not seamless.

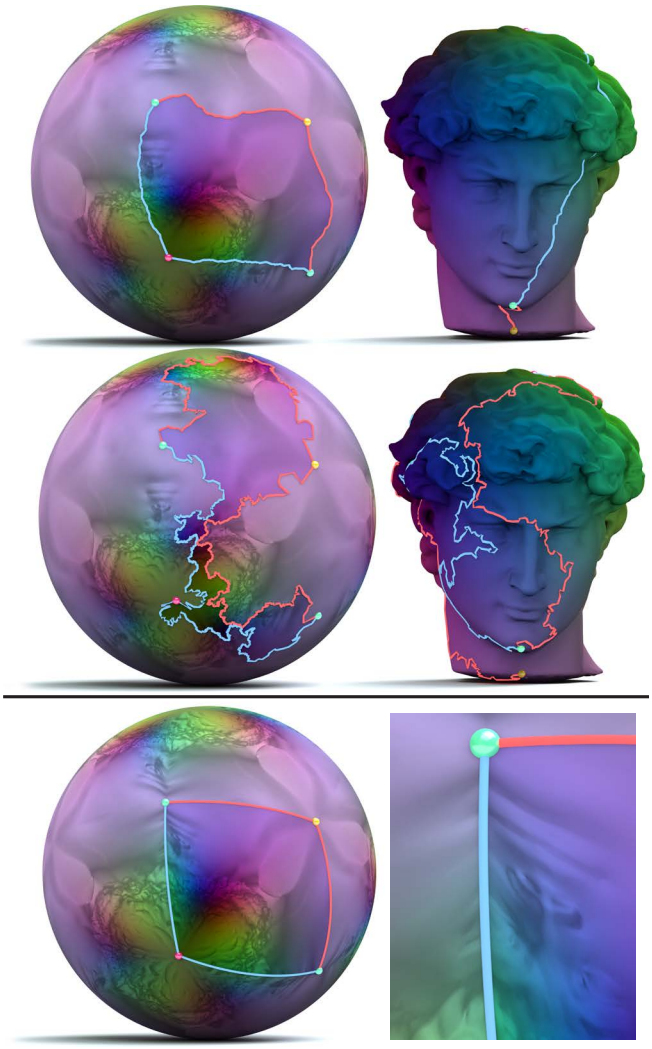


Fig. 13. Our embedding is seamless, and is only affected by the choice of cones, as shown in the top two rows: we modify the cuts on the mesh (without modifying the cone placement) and produce two embeddings – although they generate two different tiles (in bold color, left column), after tiling the results are identical and in fact represent the same orbifold embedding. In the bottom row, we show the embedding we initialize our optimization with, which, as can be seen in the blowup, *is* affected by the placement of the cuts.

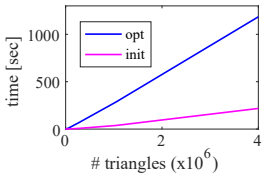
Conformality. The spherical orbifold Tutte embeddings approximate a conformal map, for the same arguments as in [Aigerman and Lipman 2015]: they are bijective minimizers of the spherical Dirichlet energy into a target domain with fixed area – the spherical orbifold – with only 2-3 points constrained to place. Figure 14 exhibits the conformality of the embeddings: we texture the meshes by projecting the spherical embedding to the plane using a stereographic projection (which preserves conformality) and then use the plane coordinates as UV coordinates. Evidently, the embeddings do not introduce almost any angle distortion.



Fig. 14. The conformality of the embeddings: texturing the models using our embeddings shows they approximate conformal maps.

6.5 Technical details

The algorithm was implemented in MATLAB. The computations were performed on a single thread on a 3.50GHz Intel i7 CPU. Typical timings of our algorithm are detailed in Table 1. Some parts of the initialization, such as the cutting, can be optimized for speed. The inset demonstrates the scalability of our algorithm; we show the initialization and optimization times for an iteratively-refined mesh, using the same cone placements.



name	# Vert	# Tri	init	# iter	opt
Moon	7K	12.5K	0.25	25	0.86
Fandisk	20K	40K	0.80	20	2.72
David	25K	50K	0.97	21	3.66
Bunny	35K	70K	1.65	19	6.12
Dragon	50K	100K	2.04	31	9.09
Armadillo	172K	346K	8.67	24	24.79
Ramesses	193K	386K	14.89	23	33.82
Frog	196K	392K	13.50	21	29.66
Lucy	263K	526K	15.05	30	61.56
Nicola	501K	1M	27.99	22	87.53
Crab	1M	2M	66.95	24	295.42

Table 1. Timings of our algorithm (initialization and optimization, measured in seconds), and number of iterations to convergence.

7 CONCLUSION

The spherical orbifold Tutte embedding is a fast and robust method for computing seamless, bijective embeddings into spherical cone domains. They are conformal parameterizations with cone types which do not exist in the Euclidean setting, and hence enable producing conformal maps which are more isometric, *i.e.*, have lower area-distortion than their Euclidean equivalents.

One limitation of our method is that our theoretical guarantees for the orbifold mapping still require two extra assumptions. The experiments strongly imply that these assumptions always hold, and we mark proving this conjecture as important future theoretical

work. A second limitation is that we can control the mapping only by prescribing the cone vertices and their angles; designing an algorithm to embed into general spherical cone-manifolds would be an interesting followup. Another question regarding the spherical embeddings is how to automatically choose the cones' location and order (*e.g.*, Figure 7) to achieve minimal isometric distortion.

In a broader sense, this work completes the construction of orbifold Tutte embeddings to all three classic geometries – Euclidean, hyperbolic and spherical; as these three geometries arise in many different mathematical contexts, we are sure the ability to compute bijective embeddings into all three will prove to be beneficial. One immediate and interesting followup that can harness the theory and practice developed here is to consider spaces of *bounded curvature*; it stands to reason that for some bound, these domains will also enable bijective Tutte embeddings, and will be much more flexible.

ACKNOWLEDGMENTS

This research was supported in part by the European Research Council (ERC Starting Grant "Surf-Comp", Grant No. 307754), I-CORE program of the Israel PBC and ISF (Grant No. 4/11) and the Simons Foundation Math+X Investigator award. The Einstein model is from Pinshape, and the crab is from Turbosquid. All other meshes are from the aim@shape repository.

REFERENCES

Noam Aigerman and Yaron Lipman. 2015. Orbifold Tutte Embeddings. *ACM Trans. Graph.* 34, 6, Article 190 (Oct. 2015), 12 pages. DOI: https://doi.org/10.1145/2816795.2818099

Noam Aigerman and Yaron Lipman. 2016. Hyperbolic orbifold tutte embeddings. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 217.

Marc Alexa. 1999. Merging polyhedral shapes with scattered features. In *Shape Modeling and Applications, 1999. Proceedings. Shape Modeling International'99. International Conference on.* IEEE, 202–210.

Arul Asirvatham, Emil Praun, and Hugues Hoppe. 2005. Consistent spherical parameterization. In *International Conference on Computational Science.* Springer, 265–272.

David Bommes, Henrik Zimmer, and Leif Kobbelt. 2009. Mixed-integer Quadrangulation. *ACM Trans. Graph.* 28, 3, Article 77 (July 2009), 10 pages. DOI: https://doi.org/10.1145/1531326.1531383

Samuel R Buss and Jay P Fillmore. 2001. Spherical averages and applications to spherical splines and interpolation. *ACM Transactions on Graphics (TOG)* 20, 2 (2001), 95–126.

Edward Chien, Zohar Levi, and Ofir Weber. 2016. Bounded distortion parametrization in the space of metrics. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 215.

John Horton Conway, Heidi Burgiel, and Chaim Goodman-Strauss. 2008. *The symmetries of things.* A.K. Peters, Wellesley (Mass.). http://opac.inria.fr/record=b1130158

Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Robust Fairing via Conformal Curvature Flow. *ACM Trans. Graph.* 32, 4 (2013).

Michael Floater. 2003. One-to-one piecewise linear mappings over triangulations. *Math. Comp.* 72, 242 (2003), 685–696.

Ilja Friedel, Peter Schröder, and Mathieu Desbrun. 2007. Unconstrained spherical parameterization. *Journal of Graphics, GPU, and Game Tools* 12, 1 (2007), 17–26.

Steven J Gortler, Craig Gotsman, and Dylan Thurston. 2006. Discrete one-forms on meshes and applications to 3D mesh parameterization. *Computer Aided Geometric Design* 23, 2 (2006), 83–112.

Craig Gotsman, Xianfeng Gu, and Alla Sheffer. 2003. Fundamentals of spherical parameterization for 3D meshes. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 358–363.

Xianfeng Gu, Yalin Wang, Tony F Chan, Paul M Thompson, and Shing-Tung Yau. 2003. Genus zero surface conformal mapping and its application to brain surface mapping. In *Biennial International Conference on Information Processing in Medical Imaging.* Springer, 172–184.

Steven Haker, Sigurd Angenent, Allen Tannenbaum, Ron Kikinis, Guillermo Sapiro, and Michael Halle. 2000. Conformal Surface Parameterization for Texture Mapping. *IEEE Transactions on Visualization and Computer Graphics* 6, 2 (April 2000), 181–189. DOI: https://doi.org/10.1109/2945.856998

Kai Hormann, Bruno Lévy, and Alla Sheffer. 2007. Mesh Parameterization: Theory and Practice Video Files Associated with This Course Are Available from the Citation Page. In *ACM SIGGRAPH 2007 Courses (SIGGRAPH '07).* ACM, New York, NY, USA, Article 1. DOI: https://doi.org/10.1145/1281500.1281510

Felix Kälberer, Matthias Nieser, and Konrad Polthier. 2007. QuadCover - Surface Parameterization using Branched Coverings. *Comput. Graph. Forum*, 375–384.

Michael Kazhdan, Jake Solomon, and Mirela Ben-Chen. 2012. Can Mean-Curvature Flow be Modified to be Non-singular?. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 1745–1754.

Shahar Z. Kovalsky, Meirav Galun, and Yaron Lipman. 2016. Accelerated Quadratic Proxy for Geometric Optimization. *ACM Trans. Graph.* 35, 4, Article 134 (July 2016), 11 pages. DOI: https://doi.org/10.1145/2897824.2925920

Tiantian Liu, Sofien Bouaziz, and Ladislav Kavan. 2016. Towards Real-time Simulation of Hyperelastic Materials. *arXiv preprint arXiv:1604.07378* (2016).

László Lovász. 2004. Discrete analytic functions: an exposition. *Surveys in differential geometry* 9 (2004), 241–273.

László Lovász and Alexander Schrijver. 1999. On the null space of a Colin de Verdiere matrix. In *Annales de l'institut Fourier*, Vol. 49. 1017–1026.

Ashish Myles and Denis Zorin. 2012. Global Parametrization by Incremental Flattening. *ACM Trans. Graph.* 31, 4, Article 109 (July 2012), 11 pages. DOI: https://doi.org/10.1145/2185520.2185605

Ashish Myles and Denis Zorin. 2013. Controlled-distortion Constrained Global Parametrization. *ACM Trans. Graph.* 32, 4, Article 105 (July 2013), 14 pages. DOI: https://doi.org/10.1145/2461912.2461970

Jorge Nocedal and Stephen Wright. 2006. *Numerical optimization.* Springer Science & Business Media.

Ulrich Pinkall and Konrad Polthier. 1993. Computing Discrete Minimal Surfaces and Their Conjugates. *Experimental Mathematics* 2 (1993), 15–36.

Emil Praun and Hugues Hoppe. 2003. Spherical parametrization and remeshing. In *ACM Transactions on Graphics (TOG)*, Vol. 22. ACM, 340–349.

Nicolas Ray, Wan Chiu Li, Bruno Lévy, Alla Sheffer, and Pierre Alliez. 2006. Periodic Global Parameterization. *ACM Trans. Graph.* 25, 4 (Oct. 2006), 1460–1485. DOI: https://doi.org/10.1145/1183287.1183297

Shadi Saba, Irad Yavneh, Craig Gotsman, and Alla Sheffer. 2005. Practical spherical embedding of manifold triangle meshes. In *International Conference on Shape Modeling and Applications 2005 (SMI'05).* IEEE, 256–265.

Avner Shapiro and Ayellet Tal. 1998. Polyhedron realization for shape transformation. *The Visual Computer* 14, 8 (1998), 429–444.

Alla Sheffer, Craig Gotsman, and Nira Dyn. 2004. Robust spherical parameterization of triangular meshes. In *Geometric Modelling.* Springer, 185–193.

Alla Sheffer, Emil Praun, and Kenneth Rose. 2006. Mesh Parameterization Methods and Their Applications. *Found. Trends. Comput. Graph. Vis.* 2, 2 (Jan. 2006), 105–171. DOI: https://doi.org/10.1561/0600000011

Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal equivalence of triangle meshes. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 77.

Y. Tong, P. Alliez, D. Cohen-Steiner, and M. Desbrun. 2006. Designing Quadrangulations with Discrete Harmonic Forms. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing (SGP '06).* Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 201–210. http://dl.acm.org/citation.cfm?id=1281957.1281983

Alex Tsui, Devin Fenton, Phong Vuong, Joel Hass, Patrice Koehl, Nina Amenta, David Coeurjolly, Charles DeCarli, and Owen Carmichael. 2013. Globally Optimal Cortical Surface Matching with Exact Landmark Correspondence. In *Proceedings of the 23rd International Conference on Information Processing in Medical Imaging (IPMI'13).* Springer-Verlag, Berlin, Heidelberg, 487–498. DOI: https://doi.org/10.1007/978-3-642-38868-2_41

William T Tutte. 1963. How to draw a graph. *Proc. London Math. Soc* 13, 3 (1963), 743–768.

Chunxue Wang, Zheng Liu, and Ligang Liu. 2014. As-rigid-as-possible spherical parametrization. *Graphical Models* 76, 5 (2014), 457–467.

Ofir Weber and Denis Zorin. 2014. Locally injective parametrization with arbitrary fixed boundaries. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 75.

APPENDIX A CONSTRUCTION OF THE BOUNDARY CONDITIONS

Preliminaries. First, we review a simple algorithm for constructing a spherical triangle having with 3 given angles, $\theta_1, \theta_2, \theta_3$. To this end, we first infer the lengths of each of the arcs of the triangle, denoted as $\Theta_1, \Theta_2, \Theta_3$, measured in radians. The arc-lengths can be computed by the well-known spherical-trigonometry formula

$$\Theta_i = \operatorname{acos}\left(\frac{\cos(\theta_i) + \cos(\theta_{i+1})\cos(\theta_{i+2})}{\sin(\theta_{i+1})\sin(\theta_{i+1})}\right),$$

where the indices are cyclically shifted. Now we compute the position of the triangle's vertices, p_1, p_2, p_3: we set p_1 to the north pole,

p_2 rotated from p_1 on the y-z plane by Θ_2, and p_3 rotated by Θ_3 on the y-z plane and then rotated by θ_1 on the x-y plane:

$$\begin{aligned} p_1 &= (0,0,1) \\ p_2 &= (0,\sin(\Theta_2),\cos(\Theta_2)) \\ p_3 &= (\sin(\theta_1)\sin(\Theta_3), cos(\theta_1)sin(\Theta_3),\cos(\Theta_3)) \,. \end{aligned} \tag{12}$$

Now we can discuss the construction of the boundary conditions for orbifolds with disk topology, sphere topology, and 2-coned orbifolds.

Disk topology. Given the order of each of the 3 reflective cones, n_1, n_2, n_3, we set the 3 angles $\alpha_1, \alpha_2, \alpha_3$ according to $\alpha_i = \frac{2\pi}{2n_i}$. Using formula (12) we construct a spherical triangle having those angles. This is a basic tile of the corresponding orbifold. The reflection matrix R_{ij}, associated to the side (p_i, p_j) of the tile, is defined as the (unique) reflection satisfying $R_{ij}p_i = p_i, R_{ij}p_j = p_j$.

Sphere topology. In this case, given the order of each of the 3 cones, we construct the same triangle as in the previous paragraph. Denoting this triangle as q_1, q_2, q_3, we add the reflected point $q_4 = (-q_3^x, q_3^y, q_3^z)$ (superscript referring to coordinate indices). We now build the quadrilateral (p_1, p_2, p_3, p_4) according to $p_1 = q_1, p_2 = q_4, p_3 = q_2, p_4 = q_3$. The 4 vertices are arranged in order, and the two pairs of matching edges are $e_1 = (p_1, p_2) \leftrightarrow (p_1, p_4) = e_4$, and $e_2 = (p_2, p_3) \leftrightarrow (p_4, p_3) = e_3$. The two rotations $\mathcal{R}_1, \mathcal{R}_2$ are defined as the 3D rotations mapping corresponding edges to one-another: $\mathcal{R}_1(e_1) = e_4, \mathcal{R}_2(e_2) = e_3$.

2-cone orbifolds. The last case is when there are only two cones. In this case, one is placed at the north pole, and one at the south pole, and the given symmetry order of the two cones is identical. In the spherical case the rotation between the two sides of the polygon is the x-y in-plane rotation by α, denoted R^α. In the disk case, we generate two vectors, $v_1 = (1,0,0)$, and $v_2 = R^\alpha v_1$. The reflection matrices $\mathcal{R}_1, \mathcal{R}_2$ associated to each of the two sides are the ones satisfying $\mathcal{R}_i v_i = -v_i$.

APPENDIX B OPTIMIZATION ALGORITHM

We note that the Euclidean Dirichlet energy provides a natural quadratic approximation to the spherical Dirichlet energy (2), as for points sufficiently close on the sphere

$$E(\Phi) \approx \sum_{(i,j)\in\mathbf{E}} w_{ij} \left\|\Phi_i - \Phi_j\right\|_2^2 = x^T (\mathcal{L} \otimes I_3)\, x,$$

where $x = \mathrm{vec}(\Phi) \in \mathbb{R}^{3|\mathbf{V}|}$ is the column stack of Φ, and the Kronecker product $(\mathcal{L} \otimes I_3)$ is the cotangent Laplacian [Pinkall and Polthier 1993] acting separately on each coordinate in $\mathbb{R}^3$. We therefore follow [Liu et al. 2016] and modify the L-BFGS two-loop recursion (see Algorithm 7.4 in [Nocedal and Wright 2006]) so that it uses a scaled Laplacian as an initial Hessian approximation. We further modify the L-BFGS algorithm to handle the linear constraints (6c) by introducing a change of coordinates, realized by a matrix $\mathcal{N}$ whose columns form an orthonormal basis for the null space of the corresponding linear system. The modified L-BFGS algorithm, described in Algorithm 1, is then iteratively used with backtracking (Armijo) line search to minimize the energy until convergence.

Our optimization is implemented in MATLAB. We set $m = 3$ in Algorithm 1. The matrix $\mathcal{N}$ is computed with a QR decomposition. An LU decomposition is used to prefactorize the matrix $\mathcal{N}^T(\mathcal{L} \otimes I_3)\mathcal{N}$, in turn enabling efficient computation of step 8 of Algorithm 1 using back-substitution.

Initialization. We initialize the optimization with a feasible starting point, by embedding the mesh into the basic tile of the desired orbifold. For each boundary curve, *e.g.*, $\gamma_i^{\blacktriangleleft}$, we place its vertices equidistantly along the corresponding arc of the basic tile. We take the plane H having the centroid of the basic tile as its normal and rotate the sphere so H is the $x-y$ plane. We project all boundary vertices onto the plane, $(x,y,z) \to (x,y,0)$; the boundary projection forms a convex polygon in the plane, which we use as fixed boundary conditions for solving the classic, Euclidean Tutte embedding, which we then project back onto the sphere via the inverse projection $(x,y) \to (x,y,\sqrt{1-x^2-y^2})$ and rotate the sphere back so H lies at its original place. This yields an initial feasible point for problem (6).

Algorithm 1: Modified L-BFGS two-loop recursion

1 $q \leftarrow \mathcal{N}^T \nabla E_k$;
2 **for** $i = k-1, k-2, \ldots, k-m$ **do**
3 $\quad s_i \leftarrow \mathcal{N}^T(x_{i+1} - x_i)$;
4 $\quad y_i \leftarrow \mathcal{N}^T(\nabla E_{i+1} - \nabla E_i)$;
5 $\quad \rho_i \leftarrow 1/(y_i^T s_i)$;
6 $\quad \alpha_i \leftarrow \rho_i s_i^T q$;
7 $\quad q \leftarrow q - \alpha_i y_i$;
8 $r \leftarrow \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}} \left(\mathcal{N}^T(\mathcal{L} \otimes I_3)\mathcal{N}\right)^{-1} q$;
9 **for** $i = k-1, k-2, \ldots, k-m$ **do**
10 $\quad \beta \leftarrow \rho_i y_i^T r$;
11 $\quad r \leftarrow r + s_i(\alpha_i - \beta)$;
12 **return** $\mathcal{N} r$;

APPENDIX C PROOF OF LEMMAS

Proof of Lemma 1. As noted in [Buss and Fillmore 2001], if $x, y \in S^2$ lie on opposite sides of a plane A passing through the origin and R_A is the reflection preserving A, then $d(x, R_A(y)) \le d(x,y)$, where the equality case happens only if either x or y lie on A.

Let Φ denote an embedding of $\mathbf{M}$ where not all vertices are contained in H. Denote by A the plane that contains ∂H, and R_A the corresponding reflection. Reflecting all $\Phi_i \notin H$ via R_A does not increase the Dirichlet energy $E(\Phi)$. Since the sphere is compact it follows that there exists a minimum where Φ is contained in H. □

proof of Lemma 2. Note that the spherical convex polygon Ω is the intersection of H with a collection of half-spaces $\{B_i\}_{i=1}^n$ that contain the origin, $\Omega = \bigcap_{i=1}^n (B_i \bigcap H)$; the gnomonic projection maps each $B_i \bigcap H$ to a 2-dimensional half-space, bijectively. Since the intersection operator commutes with bijective transformations, we get $g(\Omega) = g\left(\bigcap_{i=1}^n B_i \bigcap H\right) = \bigcap_{i=1}^n g(B_i \bigcap H)$ – an intersection of planar half-spaces, yielding a convex polygon in the plane. □